



Using Trino to analyze PLG user activation funnel

Mei Long | PM
mei@upsolver.com

November 2022



What does PLG mean to you?

- Acquisition
- Activation
- Satisfaction
- Retention
- Expansion

NORTH STAR METRIC

VALUE



Lay the groundwork

- Collect everything and ask questions
 - Does it provide value? Quantitative vs. Qualitative
 - Is it actionable?
- Evolving north star metric
- Data latency matters
- Decoupled and self-serviced architecture
- **Data quality, data quality, data quality!**



Upsolver classic PLG

Lessons Learned

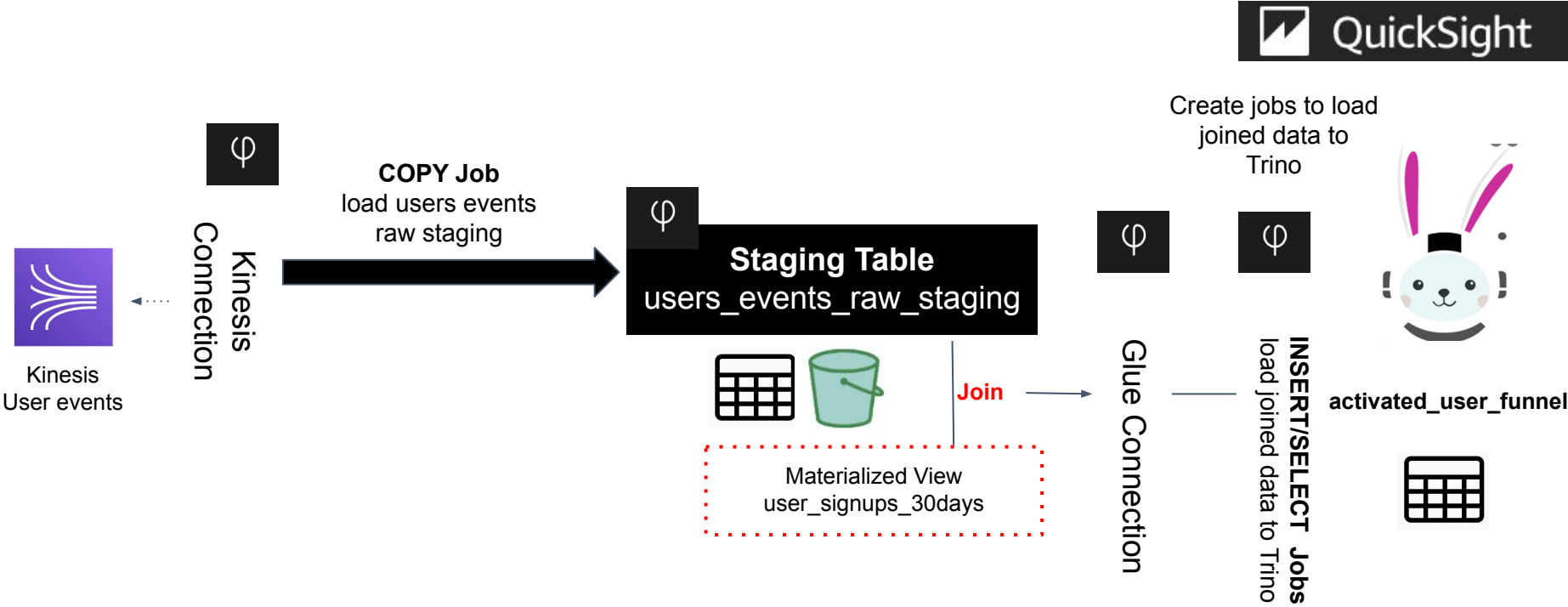
What worked for us?

- Easy *
- Handrails

What didn't work for us?

- Unanswered questions
- Inability to handle changes
- Data quality

Demo: streaming joined data to Trino



THANK YOU

Community Slack



Upsolver SQLake

<https://sqlake.upsolver.com/signup>

Upsolver SQLake: create connections

```
CREATE KINESIS CONNECTION kinesis_customer360

  AWS_ROLE =

  'arn:aws:iam::4288987458:role/upsolver-role-355e3b5b-afa7-12f4fd8327eb'

  EXTERNAL_ID = '355e3b5f-960d-406b-afa7-12f4fd8327eb'

  REGION = 'us-east-1'

  READ_ONLY = TRUE;
```

Create target Trino connection

```
CREATE S3 CONNECTION galaxy_connect
```

```
AWS_ROLE =
```

```
'arn:aws:iam::4839487947:role/upsolver-role-3djgg-960d-406b-afa7-12f4fd8327eb'
```

```
EXTERNAL_ID = '3aoiduof5f-960d-406b-afa7-12f4fd8327eb';
```

```
CREATE GLUE_CATALOG CONNECTION GALAXY
```

```
AWS_ROLE =
```

```
'arn:aws:iam::4839487947:role/upsolver-role-3djgg-960d-406b-afa7-12f4fd8327eb'
```

```
EXTERNAL_ID = '3aoiduof5f-960d-406b-afa7-12f4fd8327eb'
```

```
DEFAULT_STORAGE_CONNECTION = galaxy_connect
```

```
DEFAULT_STORAGE_LOCATION = 's3://upsolvergalaxy/'
```

```
REGION = 'us-east-1';
```


Stream raw data into staging table

```
CREATE TABLE galaxy.hive.users_events_raw_staging  
PARTITIONED BY $event_date;
```

```
CREATE JOB load_user_events_tracking_stage  
START_FROM = BEGINNING  
AS COPY FROM KINESIS kinesis_customer360 STREAM = 'user-event-tracking'  
INTO galaxy.hive.users_events_raw_staging;
```

Querying raw data from Trino near real time

Cluster explorer

- sample
- upsolver
 - hive
 - Filter schemas
 - classicmodels
 - default
 - deltalake
 - dremiotest
 - gravity
 - hive
 - activated_user_funnel
 - activated_user_funnel...
 - educated_user_funnel
 - educated_user_funnel...
 - hivetesttable
 - hivetesttable_underlyin...
 - users_events_raw_stagi...
 - users_events_raw_stagi...
 - hivedatabase
 - information_schema
 - jennytest
 - limor
 - meistagtest
 - meitagtest
 - meitest
 - northwind

Run selected (limit 1000)

```
1 SELECT * FROM "hive"."hive"."users_events_raw_staging" LIMIT 10;
```

Finished Avg. read speed 425 rows/s Elapsed time 12s Rows 10

Query details Trino UI Download

\$event_date	upsolver_schema_v...	\$commit_time	\$event_time	\$source_time	anonymous_id	api_info
2022-11-10		1 NULL	NULL	2022-11-10 15:26:06.217	d391e086-0b81-4b3e-...	{ build = { branch =
2022-11-10		1 NULL	NULL	2022-11-10 15:26:26.446	d391e086-0b81-4b3e-...	{ build = { branch =
2022-11-10		1 NULL	NULL	2022-11-10 15:34:31.759	d391e086-0b81-4b3e-...	{ build = { branch =
2022-11-10		1 NULL	NULL	2022-11-10 15:34:31.759	d391e086-0b81-4b3e-...	{ build = { branch =
2022-11-10		1 NULL	NULL	2022-11-10 15:34:40.065	4e0bf5fc-ca60-42e6-9...	{ build = { branch =
2022-11-10		1 NULL	NULL	2022-11-10 16:34:11.089	NULL	{ build = { branch =

Create materialized view and target table

```
CREATE MATERIALIZED VIEW default_glue_catalog.database_4b345d.user_signups_30days AS
  SELECT user_id, first(PARSE_DATETIME(replace(substr(event_time, 1, 23), 'T', ' '),
    'Y-MM-d HH:mm:ss.SSS')) as event_timestamp
  FROM GALAXY.hive.users_events_raw_staging
  WHERE event_name = 'sign_up'
  GROUP BY 1
  WINDOW 30 DAYS;
```

```
CREATE TABLE GALAXY.hive.activated_user_funnel(
  user_id string, event_name string, event_timestamp timestamp)
  PARTITIONED BY event_name, user_id
  GLOBALLY_UNIQUE_KEYS = TRUE;
```

Stream aggregated data into target table

```
CREATE JOB load_activated_funnel_galaxy
  START_FROM = BEGINNING
  AS INSERT INTO GALAXY.hive.activated_user_funnel MAP_COLUMNS_BY_NAME
  SELECT md5(events.user_id) as user_id
    , case when event_name = 'sign_up' then '1. Sign up'
      when lower(template_name) like '%sample data%' then '2. Launch sample data template'
      when step = 'create_connection' then '3. Create connection'
      when step = 'create_table' then '4. Create table'
      when step = 'create_copy_from_job' then '5. Copy from job'
      when step = 'create_transform_job' then '6. Transformation job'
      when step = 'congratulations' then '7. View transformation results'
    end as event_name
    , first(PARSE_DATETIME(replace(substr(event_time, 1, 23), 'T', ' '), 'Y-MM-d
HH:mm:ss.SSS')) as event_timestamp
  FROM GALAXY.hive.users_events_raw_staging AS events
```

Join with materialized view

```
LEFT JOIN default_glue_catalog.database_4b345d.user_signups_30days AS signups
  ON events.user_id = signups.user_id
WHERE $commit_time BETWEEN run_start_time() - PARSE_DURATION('30d') AND run_end_time()
AND signups.user_id is not null
AND events.user_id not like '%tstmail.link'
AND (event_name = 'sign_up'
     or lower(template_name) like '%sample data%'
     or step in ('create_connection', 'create_table', 'create_copy_from_job',
                'create_transform_job', 'congratulations'))
GROUP BY 1, 2;
```

Querying aggregated table in near real time

The screenshot shows a data platform interface with two tabs: 'Tab 1' and 'Tab 2'. The 'Cluster explorer' on the left shows a tree view with 'sample' and 'upsolver'. A green 'Run (limit 1000)' button is visible. The SQL query is:

```
1 SELECT * FROM "hive"."hive"."users_events_raw_staging" LIMIT 10;  
2  
3 SELECT * FROM "hive"."hive"."activated_user_funnel" LIMIT 10;
```

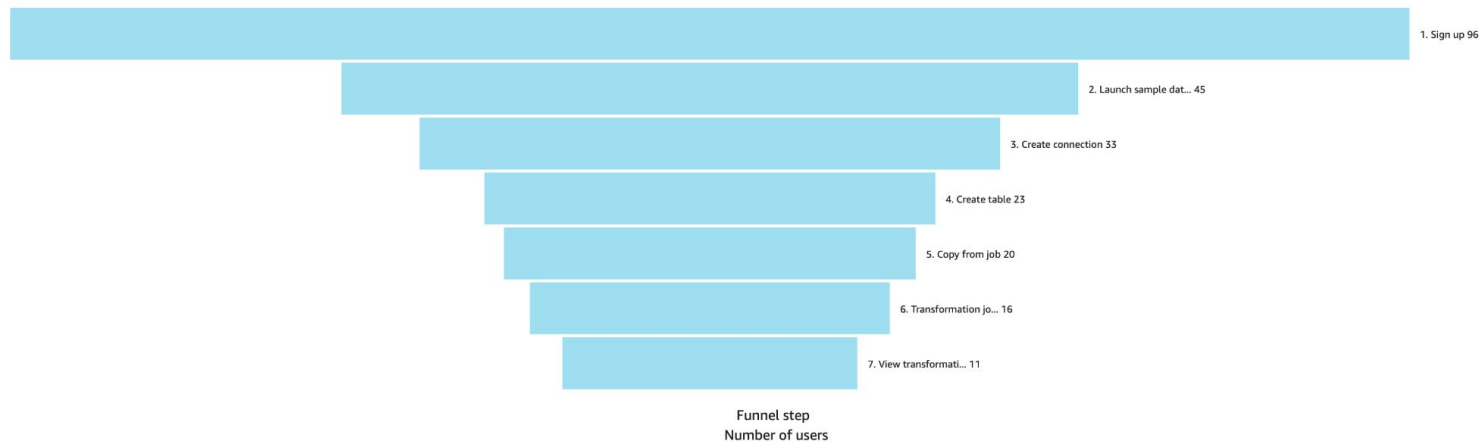
The query execution status is 'Finished' with an average read speed of 7.5K rows/s, an elapsed time of 46s, and 10 rows returned. The results table has the following columns: event_name, user_id, upsolver_schema_v..., \$commit_time, \$event_time, \$source_time, and event_timestamp. The data shows six 'Sign up' events with various user IDs and timestamps.

event_name	user_id	upsolver_schema_v...	\$commit_time	\$event_time	\$source_time	event_timestamp
1. Sign up	17d91033bb1577a94ac7...	1	NULL	NULL	2022-11-06 13:10:21.671	2022-11-06 13:10:11
1. Sign up	317fe8b099749d32b8e...	1	NULL	NULL	2022-11-06 13:48:48.209	2022-11-06 13:48:4
1. Sign up	87678b9648ce60e74a...	1	NULL	NULL	2022-11-07 10:18:55.393	2022-11-07 10:18:4
1. Sign up	02af1259af29fd1c66121...	1	NULL	NULL	2022-11-08 14:19:24.543	2022-11-08 14:19:11
1. Sign up	207dc88c93856e323f9...	1	NULL	NULL	2022-11-06 10:07:19.900	2022-11-06 10:07:1
1. Sign up	a5663d6aa7dd2b5bf7...	1	NULL	NULL	2022-11-07 06:04:51.918	2022-11-07 06:04:5

New users to 'Educated' funnel - last 30 days

Allows us to identify the friction points as new users build their first pipeline.

Users that reach the end of this funnel are considered "Educated," meaning that they learned about the building blocks that make up a pipeline and saw them in action.



Time in minutes between educated user funnel steps

To help us identify friction points for new users within the funnel

Funnel step	1. Sign up	2. Launch sample data template	3. Create connection	4. Create table	5. Copy from job	6. Transformation job	7. View transformation results
Min	0	0.25	0.17	0.13	0.24	1.99	1.7
25th percentile	0	1.07	0.33	1.02	0.87	12.89	1.99
Median	0	2.34	0.9	3.68	1.3	13.74	7.05
75th percentile	0	4.39	1.98	4.36	1.99	13.91	7.39